

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2002年 8月30日

出 願 番 号

Application Number:

特願2002-255257

[ST.10/C]:

[JP 2002-255257]

出 願 人

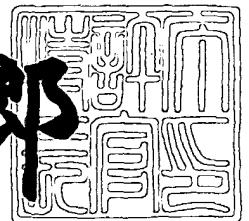
Applicant(s):

セイコーエプソン株式会社

2003年 6月16日

特 許 庁 長 官
Commissioner,
Japan Patent Office

太田信一郎



出証番号 出証特2003-3046990

【書類名】 特許願
【整理番号】 J0092593
【あて先】 特許庁長官殿
【国際特許分類】 G09G 5/24
G09G 5/26

【発明者】

【住所又は居所】 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内

【氏名】 桃蘭 幸信

【発明者】

【住所又は居所】 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内

【氏名】 胡桃澤 孝

【特許出願人】

【識別番号】 000002369

【氏名又は名称】 セイコーエプソン株式会社

【代理人】

【識別番号】 100095728

【弁理士】

【氏名又は名称】 上柳 雅誉

【連絡先】 0 2 6 6 - 5 2 - 3 1 3 9

【選任した代理人】

【識別番号】 100107076

【弁理士】

【氏名又は名称】 藤網 英吉

【選任した代理人】

【識別番号】 100107261

【弁理士】

【氏名又は名称】 須澤 修

【手数料の表示】

【予納台帳番号】 013044

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0109826

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 フォント処理装置、端末装置、フォント処理方法およびフォント処理プログラム

【特許請求の範囲】

【請求項 1】 ビットマップフォントのフォントデータを取得するデータ取得手段と、

前記フォントデータの画素構成を縦方向に分析する縦方向分析手段と、

縦方向の分析結果に基づいて、前記フォントデータを横方向に拡大又は縮小する横方向拡大／縮小手段と、

前記フォントデータの画素構成を横方向に分析する横方向分析手段と、

横方向の分析結果に基づいて、前記フォントデータを縦方向に拡大又は縮小する縦方向拡大／縮小手段と、を備えることを特徴とするフォント処理装置。

【請求項 2】 前記縦方向分析手段は、

前記フォントデータを複数の行に分割する分割手段と、

前記行ごとに、画素構成の特徴を示すコストを計算する計算手段と、を備え、

前記横方向拡大／縮小手段は、前記コストの小さい行から順に、所定数の行について画素の拡大又は縮小処理を行うことを特徴とする請求項 1 に記載のフォント処理装置。

【請求項 3】 前記縦方向分析手段は、

前記フォントデータを複数の行に分割する分割手段と、

前記行ごとに、画素構成の特徴を示すコストを計算する計算手段と、を備え、

前記横方向拡大／縮小手段は、前記コストの大きい行から順に、所定数の行について画素の拡大又は縮小処理を行うことを特徴とする請求項 1 に記載のフォント処理装置。

【請求項 4】 前記横方向分析手段は、

前記フォントデータを複数の列に分割する分割手段と、

前記列ごとに、画素構成の特徴を示すコストを計算する計算手段と、を備え、

前記縦方向拡大／縮小手段は、前記コストの小さい列から順に、所定数の列について画素の拡大又は縮小処理を行うことを特徴とする請求項 1 乃至 3 のいずれ

か一項に記載のフォント処理装置。

【請求項 5】 前記横方向分析手段は、
前記フォントデータを複数の列に分割する分割手段と、
前記列ごとに、画素構成の特徴を示すコストを計算する計算手段と、を備え、
前記縦方向拡大／縮小手段は、前記コストの大きい列から順に、所定数の列について画素の拡大又は縮小処理を行うことを特徴とする請求項 1 乃至 3 のいずれか一項に記載のフォント処理装置。

【請求項 6】 ビットマップフォントのフォントデータを取得するデータ取得手段と、

前記フォントデータの横方向拡大処理を行う横方向拡大処理手段と、
前記フォントデータの縦方向拡大処理を行う縦方向拡大処理手段と、を備え、
前記横方向拡大処理手段は、
前記フォントデータを複数の行に分割する手段と、
前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、
前記コストの小さい行から順に、所定数の行を横方向に拡大する手段と、を備え、

前記縦方向拡大処理手段は、
前記フォントデータを複数の列に分割する手段と、
前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、
前記コストの小さい列から順に、所定数の列を縦方向に拡大する手段と、を備えることを特徴とするフォント処理装置。

【請求項 7】 ビットマップフォントのフォントデータを取得するデータ取得手段と、

前記フォントデータの横方向拡大処理を行う横方向拡大処理手段と、
前記フォントデータの縦方向拡大処理を行う縦方向拡大処理手段と、を備え、
前記横方向拡大処理手段は、
前記フォントデータを複数の行に分割する手段と、
前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、
前記コストの大きい行から順に、所定数の行を横方向に拡大する手段と、を備

え、

前記縦方向拡大処理手段は、

前記フォントデータを複数の列に分割する手段と、

前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、

前記コストの大きい列から順に、所定数の列を縦方向に拡大する手段と、を備えることを特徴とするフォント処理装置。

【請求項 8】 前記所定数は、前記フォントデータの縦方向及び横方向の画素数と、前記フォントデータの拡大率とにより決定されることを特徴とする請求項 6 又は 7 に記載のフォント処理装置。

【請求項 9】 前記コスト計算手段は、

前記行又は列ごとに、当該行又は列を構成する画素数をコストとして計算する手段と、

当該行又は列の方向に隣接画素を有する画素数を前記コストに加算する手段と、を備えることを特徴とする請求項 6 乃至 8 のいずれか一項に記載のフォント処理装置。

【請求項 10】 前記コスト計算手段は、

前記行又は列ごとに、当該行又は列を構成する画素数を計算する手段と、

当該行又は列の方向に隣接画素を有する画素数を前記当該行又は列を構成する画素数に加算し、加算された結果の逆数をコストとして計算する手段と、を備えることを特徴とする請求項 6 乃至 8 のいずれか一項に記載のフォント処理装置。

【請求項 11】 前記フォントデータは、当該フォントデータの拡大処理において縦方向拡大処理と横方向拡大処理を実行すべき順序を示す処理順序情報を含み、

前記処理順序情報に従って、前記横方向拡大処理と前記縦方向拡大処理の実行順序を制御する順序制御手段をさらに備えることを特徴とする請求項 6 乃至 9 のいずれか一項に記載のフォント処理装置。

【請求項 12】 ビットマップフォントのフォントデータを取得するデータ取得手段と、

前記フォントデータの横方向縮小処理を行う横方向縮小処理手段と、

前記フォントデータの縦方向縮小処理を行う縦方向縮小処理手段と、を備え、
前記横方向縮小処理手段は、
前記フォントデータを複数の行に分割する手段と、
前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコスト計算手段と、
前記コストの小さい行から順に、所定数の行を横方向に縮小する手段と、を備え、
前記縦方向縮小処理手段は、
前記フォントデータを複数の列に分割する手段と、
前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコスト計算手段と、
前記コストの小さい列から順に、所定数の列を縦方向に縮小する手段と、を備えることを特徴とするフォント処理装置。

【請求項 1 3】 ビットマップフォントのフォントデータを取得するデータ取得手段と、

前記フォントデータの横方向縮小処理を行う横方向縮小処理手段と、
前記フォントデータの縦方向縮小処理を行う縦方向縮小処理手段と、を備え、
前記横方向縮小処理手段は、
前記フォントデータを複数の行に分割する手段と、
前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコスト計算手段と、
前記コストの大きい行から順に、所定数の行を横方向に縮小する手段と、を備え、
前記縦方向縮小処理手段は、
前記フォントデータを複数の列に分割する手段と、
前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコスト計算手段と、
前記コストの大きい列から順に、所定数の列を縦方向に縮小する手段と、を備えることを特徴とするフォント処理装置。

【請求項 1 4】 前記所定数は、前記フォントデータの縦方向及び横方向の画素数と、前記フォントデータの縮小率とにより決定されることを特徴とする請求項 1 2 又は 1 3 に記載のフォント処理装置。

【請求項 1 5】 前記コスト計算手段は、前記行又は列ごとに、当該行又は列の画素構成と、当該行又は列に隣接する行又は列の画素構成との排他的論理和に基づいてコストを計算することを特徴とする請求項 1 2 乃至 1 4 のいずれか一項に記載のフォント処理装置。

【請求項 1 6】 前記フォントデータは、当該フォントデータの縮小処理において縦方向縮小処理と横方向縮小処理を実行すべき順序を示す処理順序情報を含み、

前記処理順序情報に従って、前記横方向縮小処理と前記縦方向縮小処理の実行順序を制御する順序制御手段をさらに備えることを特徴とする請求項 1 2 乃至 1 5 のいずれか一項に記載のフォント処理装置。

【請求項 1 7】 前記請求項 1 乃至 1 6 のいずれか一項のフォント処理装置と、

前記フォント処理装置により生成されたフォントデータを記憶する記憶手段と、

前記フォント処理装置により生成されたフォントデータを表示する表示部と、を備えることを特徴とする端末装置。

【請求項 1 8】 ビットマップフォントのフォントデータを取得するデータ取得工程と、

前記フォントデータの画素構成を縦方向に分析する縦方向分析工程と、

縦方向の分析結果に基づいて、前記フォントデータを横方向に拡大又は縮小する横方向拡大／縮小工程と、

前記フォントデータの画素構成を横方向に分析する横方向分析工程と、

横方向の分析結果に基づいて、前記フォントデータを縦方向に拡大又は縮小する縦方向拡大／縮小工程と、を有することを特徴とするフォント処理方法。

【請求項 1 9】 ビットマップフォントのフォントデータを取得するデータ取得工程と、

前記フォントデータの横方向拡大処理を行う横方向拡大処理工程と、
前記フォントデータの縦方向拡大処理を行う縦方向拡大処理工程と、を有し、
前記横方向拡大処理工程は、
前記フォントデータを複数の行に分割する工程と、
前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算工程と、
前記コストの小さい行から順に、所定数の行を横方向に拡大する工程と、を有し、
前記縦方向拡大処理工程は、
前記フォントデータを複数の列に分割する工程と、
前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算工程と、
前記コストの小さい列から順に、所定数の列を縦方向に拡大する工程と、を有することを特徴とするフォント処理方法。

【請求項 2 0】 ビットマップフォントのフォントデータを取得するデータ取得工程と、

前記フォントデータの横方向拡大処理を行う横方向拡大処理工程と、
前記フォントデータの縦方向拡大処理を行う縦方向拡大処理工程と、を有し、
前記横方向拡大処理工程は、
前記フォントデータを複数の行に分割する工程と、
前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算工程と、
前記コストの大きい行から順に、所定数の行を横方向に拡大する工程と、を有し、

前記縦方向拡大処理工程は、
前記フォントデータを複数の列に分割する工程と、
前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算工程と、
前記コストの大きい列から順に、所定数の列を縦方向に拡大する工程と、を有することを特徴とするフォント処理方法。

【請求項 2 1】 ビットマップフォントのフォントデータを取得するデータ取得工程と、

前記フォントデータの横方向縮小処理を行う横方向縮小処理工程と、

前記フォントデータの縦方向縮小処理を行う縦方向縮小処理工程と、を有し、
前記横方向縮小処理工程は、
前記フォントデータを複数の行に分割する工程と、
前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコスト計算工程と、
前記コストの小さい行から順に、所定数の行を横方向に縮小する工程と、を有し、
前記縦方向縮小処理工程は、
前記フォントデータを複数の列に分割する工程と、
前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコスト計算工程と、
前記コストの小さい列から順に、所定数の列を縦方向に縮小する工程と、を有することを特徴とするフォント処理方法。

【請求項 2 2】 ビットマップフォントのフォントデータを取得するデータ取得工程と、

前記フォントデータの横方向縮小処理を行う横方向縮小処理工程と、
前記フォントデータの縦方向縮小処理を行う縦方向縮小処理工程と、を有し、
前記横方向縮小処理工程は、
前記フォントデータを複数の行に分割する工程と、
前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコスト計算工程と、
前記コストの大きい行から順に、所定数の行を横方向に縮小する工程と、を有し、
前記縦方向縮小処理工程は、
前記フォントデータを複数の列に分割する工程と、
前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコスト計算工程と、
前記コストの大きい列から順に、所定数の列を縦方向に縮小する工程と、を有することを特徴とするフォント処理方法。

【請求項 2 3】 コンピュータを備える端末装置において実行されるフォント処理プログラムであって、前記コンピュータを、
ビットマップフォントのフォントデータを取得するデータ取得手段、
前記フォントデータの画素構成を縦方向に分析する縦方向分析手段、
縦方向の分析結果に基づいて、前記フォントデータを横方向に拡大又は縮小する横方向拡大／縮小手段、
前記フォントデータの画素構成を横方向に分析する横方向分析手段、
横方向の分析結果に基づいて、前記フォントデータを縦方向に拡大又は縮小する縦方向拡大／縮小手段、として機能させることを特徴とするフォント処理プログラム。

【請求項 2 4】 コンピュータを備える端末装置において実行されるフォント処理プログラムであって、前記コンピュータを、
ビットマップフォントのフォントデータを取得するデータ取得手段、
前記フォントデータの横方向拡大処理を行う横方向拡大処理手段、
前記フォントデータの縦方向拡大処理を行う縦方向拡大処理手段、として機能させ、
前記横方向拡大処理手段は、
前記フォントデータを複数の行に分割する手段と、
前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、
前記コストの小さい行から順に、所定数の行を横方向に拡大する手段と、を備え、
前記縦方向拡大処理手段は、
前記フォントデータを複数の列に分割する手段と、
前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、
前記コストの小さい列から順に、所定数の列を縦方向に拡大する手段と、を備えることを特徴とするフォント処理プログラム。

【請求項 2 5】 コンピュータを備える端末装置において実行されるフォント処理プログラムであって、前記コンピュータを、
ビットマップフォントのフォントデータを取得するデータ取得手段、

前記フォントデータの横方向拡大処理を行う横方向拡大処理手段、
前記フォントデータの縦方向拡大処理を行う縦方向拡大処理手段、として機能
させ、

前記横方向拡大処理手段は、
前記フォントデータを複数の行に分割する手段と、
前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、
前記コストの大きい行から順に、所定数の行を横方向に拡大する手段と、を備
え、

前記縦方向拡大処理手段は、
前記フォントデータを複数の列に分割する手段と、
前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、
前記コストの大きい列から順に、所定数の列を縦方向に拡大する手段と、を備
えることを特徴とするフォント処理プログラム。

【請求項 2 6】 コンピュータを備える端末装置において実行されるフォン
ト処理プログラムであって、前記コンピュータを、

ビットマップフォントのフォントデータを取得するデータ取得手段、
前記フォントデータの横方向縮小処理を行う横方向縮小処理手段、
前記フォントデータの縦方向縮小処理を行う縦方向縮小処理手段、として機能
させ、

前記横方向縮小処理手段は、
前記フォントデータを複数の行に分割する手段と、
前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコス
ト計算手段と、

前記コストの小さい行から順に、所定数の行を横方向に縮小する手段と、を備
え、

前記縦方向縮小処理手段は、
前記フォントデータを複数の列に分割する手段と、
前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコス
ト計算手段と、

前記コストの小さい列から順に、所定数の列を縦方向に縮小する手段と、を備えることを特徴とするフォント処理プログラム。

【請求項 2 7】 コンピュータを備える端末装置において実行されるフォント処理プログラムであって、前記コンピュータを、

ビットマップフォントのフォントデータを取得するデータ取得手段、

前記フォントデータの横方向縮小処理を行う横方向縮小処理手段、

前記フォントデータの縦方向縮小処理を行う縦方向縮小処理手段、として機能させ、

前記横方向縮小処理手段は、

前記フォントデータを複数の行に分割する手段と、

前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコスト計算手段と、

前記コストの大きい行から順に、所定数の行を横方向に縮小する手段と、を備え、

前記縦方向縮小処理手段は、

前記フォントデータを複数の列に分割する手段と、

前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコスト計算手段と、

前記コストの大きい列から順に、所定数の列を縦方向に縮小する手段と、を備えることを特徴とするフォント処理プログラム。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、ビットマップフォントの拡大、縮小処理に関する。

【 0 0 0 2 】

【背景技術】

携帯電話や P D A (Personal Digital Assistant) などの装置では文字などの表示にビットマップフォントが使用される。ビットマップフォントは、予め用意された画素の配列パターンにより文字や記号などを表示するものである。ベクト

ルデータの集合として文字や記号などを表示するアウトラインフォントと異なり、ビットマップフォントは単純な画素の配列パターンであるため、1文字当たりのデータ量が小さい。そのため、表示エリアの画素数が比較的少ない携帯電話やPDAなどにおいては、ビットマップフォントが使用される。

【0003】

【発明が解決しようとする課題】

表示装置上に文字や記号を表示する場合、文字の拡大や縮小が要求されることがある。携帯電話やPDAなどに使用されるビットマップフォントの場合、1文字のサイズが小さく、文字の太さが1画素で表されることも多い。そのような場合、元の文字サイズと拡大後の文字サイズとの縦横比に基づいて単純な座標変換を行う手法によりビットマップフォントを拡大すると、拡大後の文字に不自然な太さの部分ができてしまうことがある。また、同様の小さなビットマップフォントを上記のような単純な座標変換により縮小すると、縮小後に文字が潰れ、不自然な太さの線ができてたり、文字の部分同士が不適切に連結して違う文字に見えてしまうことがある。

【0004】

本発明は、以上の点に鑑みてなされたものであり、携帯電話やPDAなどで使用するサイズの小さなビットマップフォントを自然に拡大・縮小することを課題とする。

【0005】

【課題を解決するための手段】

本発明の1つの観点では、フォント処理装置において、ビットマップフォントのフォントデータを取得するデータ取得手段と、前記フォントデータの画素構成を縦方向に分析する縦方向分析手段と、縦方向の分析結果に基づいて、前記フォントデータを横方向に拡大又は縮小する横方向拡大／縮小手段と、前記フォントデータの画素構成を横方向に分析する横方向分析手段と、横方向の分析結果に基づいて、前記フォントデータを縦方向に拡大又は縮小する縦方向拡大／縮小手段と、を備える。

【0006】

また、同様の観点によるフォント処理方法は、ビットマップフォントのフォントデータを取得するデータ取得工程と、前記フォントデータの画素構成を縦方向に分析する縦方向分析工程と、縦方向の分析結果に基づいて、前記フォントデータを横方向に拡大又は縮小する横方向拡大／縮小工程と、前記フォントデータの画素構成を横方向に分析する横方向分析工程と、横方向の分析結果に基づいて、前記フォントデータを縦方向に拡大又は縮小する縦方向拡大／縮小工程と、を有する。

【 0 0 0 7 】

上記のフォント処理装置又は方法は、ビットマップフォントを所定の拡大率又は縮小率で拡大又は縮小する。まず、拡大又は縮小の対象となるビットマップフォントのフォントデータを取得する。そして、フォントデータを縦方向に分析し、その分析結果に基づいて、フォントデータを横方向に拡大又は縮小する。同様に、ビットマップフォントのフォントデータを横方向に分析し、その分析結果に基づいてフォントデータを縦方向に拡大又は縮小する。これにより、拡大／縮小の対象となるフォントデータの画素構成に応じて、それぞれ縦方向及び横方向に適切な拡大／縮小処理がなされるので、処理後のフォントデータが不自然となることを防止できる。

【 0 0 0 8 】

上記のフォント処理装置の一態様では、前記縦方向分析手段は、前記フォントデータを複数の行に分割する分割手段と、前記行ごとに、画素構成の特徴を示すコストを計算する計算手段と、を備え、前記縦方向拡大／縮小手段は、前記コストの小さい行又は大きい行から順に、所定数の行について画素の拡大又は縮小処理を行う。

【 0 0 0 9 】

このフォント処理装置によれば、フォントデータを構成する画素の行方向（縦方向）の画素構成が数値化されてコストとして算出され、その値に基づいて拡大／縮小処理が行われる。よって、画素構成の分析を単純なコスト計算により行うことができ、迅速な処理が可能となる。

【 0 0 1 0 】

上記のフォント処理装置の他の一態様では、前記横方向分析手段は、前記フォントデータを複数の列に分割する分割手段と、前記列ごとに、画素構成の特徴を示すコストを計算する計算手段と、を備え、前記縦方向拡大／縮小手段は、前記コストの小さい列又は大きい列から順に、所定数の列について画素の拡大又は縮小処理を行う。

【 0 0 1 1 】

このフォント処理装置によれば、フォントデータを構成する画素の列方向（横方向）の画素構成が数値化されてコストとして算出され、その値に基づいて拡大／縮小処理が行われる。よって、画素構成の分析を単純なコスト計算により行うことができ、迅速な処理が可能となる。

【 0 0 1 2 】

本発明の他の観点では、フォント処理装置は、ビットマップフォントのフォントデータを取得するデータ取得手段と、前記フォントデータの横方向拡大処理を行う横方向拡大処理手段と、前記フォントデータの縦方向拡大処理を行う縦方向拡大処理手段と、を備え、前記横方向拡大処理手段は、前記フォントデータを複数の行に分割する手段と、前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、前記コストの小さい行又は大きい行から順に、所定数の行を横方向に拡大する手段と、を備え、前記縦方向拡大処理手段は、前記フォントデータを複数の列に分割する手段と、前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、前記コストの小さい列又は大きい列から順に、所定数の列を縦方向に拡大する手段と、を備える。

【 0 0 1 3 】

また、同様の観点によるフォント処理方法は、ビットマップフォントのフォントデータを取得するデータ取得工程と、前記フォントデータの横方向拡大処理を行う横方向拡大処理工程と、前記フォントデータの縦方向拡大処理を行う縦方向拡大処理工程と、を有し、前記横方向拡大処理工程は、前記フォントデータを複数の行に分割する工程と、前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算工程と、前記コストの小さい行又は大きい行から順に、所定数の行を横方向に拡大する工程と、を有し、前記縦方向拡大処理工程は、前記フォ

トデータを複数の列に分割する工程と、前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算工程と、前記コストの小さい列又は大きい列から順に、所定数の列を縦方向に拡大する工程と、を有する。

【 0 0 1 4 】

上記のフォント処理装置又は方法によれば、フォントデータが複数の行に分割され、その行ごとにコストが計算される。ここで、コストは画素の線分量、即ち、その行を構成する画素が単なる点又は点の集合に近いか、線分に近いかを示す指標となる。そして、コストの小さい順又は大きい順に行を横方向に拡大する。これにより、点や点の集合に近い画素構成を有する行が優先的に拡大され、線分に近い画素構成を有する行は拡大されにくくなる。よって、文字を構成する線分が必要以上に拡大されて文字全体のバランスがくずれることがなくなり、自然な拡大がなされる。

【 0 0 1 5 】

同様に、フォントデータが複数の列に分割され、その列ごとにコストが計算される。ここでも、コストは画素の線分量を示し、コストの小さい順又は大きい順に列が縦方向に拡大される。よって、点や点の集合に近い画素構成を有する列が優先的に拡大され、線分に近い画素構成を有する列は拡大されにくくなり、自然な拡大がなされる。

【 0 0 1 6 】

上記のフォント処理装置の一態様では、前記所定数は、前記フォントデータの縦方向及び横方向の画素数と、前記フォントデータの拡大率とにより決定される。即ち、フォントの拡大率に応じて、対象となるフォントを構成する縦横の画素数に対してそれぞれいくつの画素を増加させれば良いかが決定され、その数だけフォントデータを構成する縦及び横方向の画素数が増加するように拡大が行われる。

【 0 0 1 7 】

上記のフォント処理装置の他の一態様では、前記コスト計算手段は、前記行又は列ごとに、当該行又は列を構成する画素数をコストとして計算する手段と、当該行又は列の方向に隣接画素を有する画素数を前記コストに加算する手段と、を

備える。

【 0 0 1 8 】

上記のフォント処理装置の他の一態様では、前記コスト計算手段は、前記行又は列ごとに、当該行又は列を構成する画素数を計算する手段と、当該行又は列の方向に隣接画素を有する画素数を前記当該行又は列を構成する画素数に加算し、加算された結果の逆数をコストとして計算する手段と、を備える。

【 0 0 1 9 】

これらの態様によれば、コスト計算手段は、単純な演算処理によってコストを計算するので、フォント拡大処理を簡素な構成で迅速に行うことができる。

【 0 0 2 0 】

上記のフォント処理装置のさらに他の一態様では、前記フォントデータは、当該フォントデータの拡大処理において縦方向拡大処理と横方向拡大処理を実行すべき順序を示す処理順序情報を含み、前記処理順序情報に従って、前記横方向拡大処理と前記縦方向拡大処理の実行順序を制御する順序制御手段をさらに備える。

【 0 0 2 1 】

この態様によれば、拡大の対象となるフォントごとに、縦方向拡大処理と横方向拡大処理のどちらを先に行った方がより自然な拡大フォントが得られるかが予め決定され、その順序を示す処理順序情報がフォントデータに付属している。よって、フォント処理装置は、その処理順序情報を参照して縦方向拡大処理と横方向拡大処理の実行順序を決定することにより、自然な拡大フォントが得られる。

【 0 0 2 2 】

本発明のさらに他の観点では、フォント処理装置は、ビットマップフォントのフォントデータを取得するデータ取得手段と、前記フォントデータの横方向縮小処理を行う横方向縮小処理手段と、前記フォントデータの縦方向縮小処理を行う縦方向縮小処理手段と、を備え、前記横方向縮小処理手段は、前記フォントデータを複数の行に分割する手段と、前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコスト計算手段と、前記コストの小さい行又は大きい行から順に、所定数の行を横方向に縮小する手段と、を備え、前記縦方向縮小処

理手段は、前記フォントデータを複数の列に分割する手段と、前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコスト計算手段と、前記コストの小さい列又は大きい列から順に、所定数の列を縦方向に縮小する手段と、を備える。

【 0 0 2 3 】

また、同様の観点によるフォント処理方法は、ビットマップフォントのフォントデータを取得するデータ取得工程と、前記フォントデータの横方向縮小処理を行う横方向縮小処理工程と、前記フォントデータの縦方向縮小処理を行う縦方向縮小処理工程と、を有し、前記横方向縮小処理工程は、前記フォントデータを複数の行に分割する工程と、前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコスト計算工程と、前記コストの小さい行又は大きい行から順に、所定数の行を横方向に縮小する工程と、を有し、前記縦方向縮小処理工程は、前記フォントデータを複数の列に分割する工程と、前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコスト計算工程と、前記コストの小さい列又は大きい行から順に、所定数の列を縦方向に縮小する工程と、を有する。

【 0 0 2 4 】

上記のフォント処理装置又は方法によれば、フォントデータが複数の行に分割され、その行ごとにコストが計算される。ここで、コストは隣接する行の画素構成との類似度を示す。そして、コストの小さい順又は大きい順に行を横方向に拡大する。これにより、画素構成が類似する行が隣接する部分が優先的に縮小されるので、文字全体のバランスがくずれることがなくなり、自然な縮小がなされる。

【 0 0 2 5 】

同様に、フォントデータが複数の列に分割され、その列ごとにコストが計算される。ここでも、コストは隣接する列の画素構成との類似度を示し、コストの小さい順又は大きい順に列が縦方向に縮小される。よって画素構成が類似する列が隣接する部分が優先的に縮小されるので、文字全体のバランスがくずれることがなくなり、自然な縮小がなされる。

【 0 0 2 6 】

上記のフォント処理装置の一態様では、前記所定数は、前記フォントデータの縦方向及び横方向の画素数と、前記フォントデータの縮小率とにより決定される。即ち、フォントの縮小率に応じて、対象となるフォントを構成する縦横の画素数に対してそれぞれいくつの画素を減少させれば良いかが決定され、その数だけフォントデータを構成する縦及び横方向の画素数が減少するように縮小が行われる。

【 0 0 2 7 】

上記のフォント処理装置のさらに他の一態様では、前記コスト計算手段は、前記行又は列ごとに、当該行又は列の画素構成と、当該行又は列に隣接する行又は列の画素構成との排他的論理和に基づいてコストを計算する。

【 0 0 2 8 】

この態様によれば、コスト計算手段は、単純な加算処理によってコストを計算するので、フォント拡大処理を簡素な構成で迅速に行うことができる。

【 0 0 2 9 】

上記のフォント処理装置のさらに他の一態様では、前記フォントデータは、当該フォントデータの縮小処理において縦方向縮小処理と横方向縮小処理を実行すべき順序を示す処理順序情報を含み、前記処理順序情報に従って、前記横方向縮小処理と前記縦方向縮小処理の実行順序を制御する順序制御手段をさらに備える。

【 0 0 3 0 】

この態様によれば、縮小の対象となるフォントごとに、縦方向縮小処理と横方向縮小処理のどちらを先に行った方がより自然な縮小フォントが得られるかが予め決定され、その順序を示す処理順序情報がフォントデータに付属している。よって、フォント処理装置は、その処理順序情報を参照して縦方向縮小処理と横方向縮小処理の実行順序を決定することにより、自然な縮小フォントが得られる。

【 0 0 3 1 】

本発明のさらに他の観点では、上記のフォント処理装置と、前記フォント処理装置により生成されたフォントデータを記憶する記憶手段と、前記フォント処理

装置により生成されたフォントデータを表示する表示部と、を備える端末装置を構成することができる。当該端末装置は、例えば携帯電話やPDAなどの携帯型端末装置とすることができる。本発明のフォント処理装置により、単純な演算で迅速にフォントの拡大や縮小ができるので、そのような携帯型端末装置においても自然なフォント拡大／縮小を行うことができる。

【 0 0 3 2 】

本発明のさらに他の観点では、コンピュータを備える端末装置において実行されるフォント処理プログラムは、前記コンピュータを、ビットマップフォントのフォントデータを取得するデータ取得手段、前記フォントデータの画素構成を縦方向に分析する縦方向分析手段、縦方向の分析結果に基づいて、前記フォントデータを横方向に拡大又は縮小する横方向拡大／縮小手段、前記フォントデータの画素構成を横方向に分析する横方向分析手段、横方向の分析結果に基づいて、前記フォントデータを縦方向に拡大又は縮小する縦方向拡大／縮小手段として機能させる。

【 0 0 3 3 】

本発明のさらに他の観点では、コンピュータを備える端末装置において実行されるフォント処理プログラムは、前記コンピュータを、ビットマップフォントのフォントデータを取得するデータ取得手段、前記フォントデータの横方向拡大処理を行う横方向拡大処理手段、前記フォントデータの縦方向拡大処理を行う縦方向拡大処理手段、として機能させ、前記横方向拡大処理手段は、前記フォントデータを複数の行に分割する手段と、前記行ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、前記コストの小さい行又は大きい行から順に、所定数の行を横方向に拡大する手段と、を備え、前記縦方向拡大処理手段は、前記フォントデータを複数の列に分割する手段と、前記列ごとに、画素構成の線分量を示すコストを計算するコスト計算手段と、前記コストの小さい列又は大きい列から順に、所定数の列を縦方向に拡大する手段と、を備える。

【 0 0 3 4 】

本発明のさらに他の観点では、コンピュータを備える端末装置において実行されるフォント処理プログラムは、前記コンピュータを、ビットマップフォントの

フォントデータを取得するデータ取得手段、前記フォントデータの横方向縮小処理を行う横方向縮小処理手段、前記フォントデータの縦方向縮小処理を行う縦方向縮小処理手段、として機能させ、前記横方向縮小処理手段は、前記フォントデータを複数の行に分割する手段と、前記行ごとに、隣接する行の画素構成との類似度を示すコストを計算するコスト計算手段と、前記コストの小さい行又は大きい行から順に、所定数の行を横方向に縮小する手段と、を備え、前記縦方向縮小処理手段は、前記フォントデータを複数の列に分割する手段と、前記列ごとに、隣接する列の画素構成との類似度を示すコストを計算するコスト計算手段と、前記コストの小さい列又は大きい列から順に、所定数の列を縦方向に縮小する手段と、を備える。

【 0 0 3 5 】

上記のフォント処理プログラムを、端末装置内のコンピュータで実行することにより、上記のフォント処理装置を実現することができる。

【 0 0 3 6 】

【発明の実施の形態】

以下、図面を参照して本発明の好適な実施の形態について説明する。

【 0 0 3 7 】

〔携帯端末装置の構成〕

図 1 に、本発明の実施形態にかかるビットマップフォントの拡大・縮小処理を適用した携帯端末装置の概略構成を示す。図 1 において、携帯端末装置 1 0 は、例えば携帯電話や P D A など、画像表示エリアが比較的小さい端末装置である。携帯端末装置 1 0 は、表示部 1 2 と、処理フォントメモリ 1 4 と、C P U 1 6 と、入力部 1 8 と、プログラム R O M 2 0 と、フォント R O M 2 2 と、R A M 2 4 とを備える。

【 0 0 3 8 】

表示部 2 2 は、例えば L C D (Liquid Crystal Display: 液晶表示装置) などの軽量、薄型の表示装置とすることができ、表示エリア内にビットマップフォントにより構成される文字を表示する。

【 0 0 3 9 】

入力部 1 8 は、携帯電話であれば各種の操作ボタンなど、PDA であればタッチペンなどによる接触を検出するタブレットなどにより構成することができ、ユーザが各種の指示、選択を行う際に使用される。入力部 1 8 に対して入力された指示、選択などは、電気信号に変換されて CPU 1 6 へ送られる。

【 0 0 4 0 】

プログラム ROM 2 0 は、携帯端末装置 1 0 の各種機能を実行するための各種プログラムを記憶し、特に本実施形態ではビットマップフォントの拡大・縮小プログラム（以下、「フォント拡大・縮小プログラム」と呼ぶ。）、ビットマップフォントを利用した文字の表示プログラムなどを記憶している。

【 0 0 4 1 】

フォント ROM 2 2 は、ビットマップフォントの元データ（「字母データ」とも呼ぶ。）を記憶する。なお、ビットマップフォントの元データは、例えば 1 6 × 1 6 ドットなどの、縦横比が等しいフォント（「正方フォント」とも呼ぶ。）とすることが一般的である。

【 0 0 4 2 】

RAM 2 4 は、ビットマップフォントの拡大・縮小プログラムに従ってビットマップフォントの元データを拡大・縮小処理する際に作業用メモリとして使用される。一方、処理フォントメモリ 1 4 は、ビットマップフォントの拡大・縮小プログラムによって拡大又は縮小処理により作成されたフォント（以下、「処理フォント」とも呼ぶ。）を一時的に記憶するメモリである。処理フォントメモリ 1 4 は、通常、RAM やフラッシュメモリなどにより構成することができ、携帯端末装置 1 0 が電源オフされるまで記憶内容を保持する。

【 0 0 4 3 】

CPU 1 6 は、プログラム ROM 2 0 内に記憶されている各種プログラムを実行することにより、携帯端末装置 1 0 の各種機能を実行する。特に、本実施形態では、プログラム ROM 2 0 内に記憶されている文字表示プログラムを読み出して実行することにより、文字を表示部 1 2 上に表示させる。また、同じくプログラム ROM 2 0 内に記憶されているフォント拡大・縮小プログラムを読み出して実行することにより、フォント ROM 2 2 内に記憶されているビットマップフォ

ントの元データを拡大、縮小して処理フォントを生成する。なお、CPU 16は、これら以外に各種のプログラムを実行することにより携帯端末装置10の各種機能を実現するが、それらは本発明とは直接の関連を有しないので、説明を省略する。

【0044】

〔フォント拡大処理〕

次に、本発明の特徴部分であるフォント拡大・縮小処理について説明する。フォント拡大・縮小処理は、前述のようにCPU 16がプログラムROM 20内に格納されているフォント拡大・縮小プログラムを実行することにより行われる。以下、フォント拡大処理とフォント縮小処理について、順に説明する。

【0045】

まず、フォント拡大処理について図2乃至図7を参照して説明する。図2乃至図4はフォント拡大処理のメインルーチン及びサブルーチンのフローチャートであり、図5乃至図7はフォント拡大処理による各工程を説明するための図である。

【0046】

図2を参照すると、まずCPU 1はフォント拡大指示を受け取る（ステップS1）。フォント拡大指示は、例えばユーザが入力部18を操作して表示フォントのサイズの指定、選択や表示モードの指定、選択を行った場合に、入力部18からの指示に基づいて生成される。なお、ユーザによる指示以外の場合でも、表示部12上への表示内容に応じてフォントの拡大が必要となる場合もある。例えば、表示エリア内に特定の文字データを表示する際に、文字表示プログラムが自動的にフォントの拡大を自動的に要求する場合もある。

【0047】

フォントの拡大指示がなされると、CPU 1は表示対象となるフォントの元データをフォントROM 22から読み出し、作業メモリであるRAM 24へ展開する（ステップS2）。そして、CPU 1は行方向拡大処理を実行して、対象となるフォントをまず行方向に拡大し（ステップS3）、次に列方向拡大処理を実行して、行方向に拡大されたフォントをさらに列方向に拡大する（ステップS4）

。なお、本例の場合、行方向の拡大率と列方向の拡大率は同一に設定される。こうして、行方向及び列方向にフォントの拡大が完了すると、CPU 1 は拡大後のフォントデータを処理フォントとして処理フォントメモリ 1 4 に一時的に保存し（ステップ S 5）、さらに表示部 1 2 上に表示する（ステップ S 6）。こうして、特定のフォントが拡大されて表示部 1 2 上に表示される。

【 0 0 4 8 】

次に、ステップ S 3 で行われる行方向拡大処理の詳細について図 3 のフローチャートを参照して説明する。図 3 において、まず CPU 1 は、対象となるフォントを複数の行に分割する（ステップ S 1 1）。そして、CPU 1 は、分割により得られた行ごとに、コストを計算する（ステップ S 1 2）。ここで、コストとは、フォントを構成する画素の数及び隣接する画素の有無などにより求められる値であり、フォント拡大処理においては線分量を示す。即ち、コストが大きいほど、その行には点や点の集合ではなく、線分が含まれている度合いが高いことを意味する。

【 0 0 4 9 】

図 4 に、ステップ S 1 2 におけるコスト計算処理の詳細を示す。コスト計算処理では、まず、図 5（a）に示すように、行単位でフォント構成要素の数をカウントする（ステップ S 2 1）。図 5（a）において、各行の下に示した数字（左から、0、4、3、3、…）が各行のコストを示している。

【 0 0 5 0 】

次に、図 5（b）に示すように、隣接するフォント構成画素を有する画素分の加算を行う（ステップ S 2 2）。図 5（b）の例では、隣接するフォント構成画素を有するフォント構成画素ごとに、コストに「2」を加算している。こうして、隣接画素分のコストを加算した結果得られる各行のコストが各行の下に示されている。隣接する画素を有する画素についてコストを加算することにより、その行に存在するフォント構成画素が点であるのか、線分に近いのかをある程度区分することができる。即ち、1つの行に含まれるフォント構成画素数が同じであっても、コストが大きい行ほど、その行に含まれるフォント構成画素は線分に近いということが言える。そして、後述のように、コスト値の大きい行、つまり線分

に近い行ほど拡大されにくくする。

【 0 0 5 1 】

次に、図 6 (a) に示すように、スペースを考慮した重み付け処理を行う（ステップ S 2 3）。具体的には、各行について、両端の画素がスペース（フォント構成画素が存在しない。図 5 及び 6 ではスペースで示されている。）である場合には、コスト値を「+ 2」する。また、両端の画素の 1 画素内側の画素がスペースである場合、コスト値を「+ 1」する。この処理は、文字としてのバランス及び文字を横方向に並べて文にした場合のバランスの面を考慮している。例えば平仮名の「し」という文字などは、普通に拡大すると左右の空きの部分だけが拡大されてしまい、文字を構成する中央のフォント構成画素はあまり拡大されない。その結果、拡大後の文章として見ると、他の文字は拡大されているが、文字「し」だけはフォント構成部分が十分に拡大されていないため、小さいままに見えてしまう。このような不具合を防止するために、スペースを考慮した重み付けを行うのである。この処理の結果、上下端にスペースを有する行はコスト値が大きくなり、拡大されにくくなる。

【 0 0 5 2 】

次に、同一コスト値を有する行の重み付け処理を行う（ステップ S 2 4）。具体的には、ある行について、隣接する行とフォント構成画素のパターンを比較し、隣接する行とフォント構成画素のパターンが同じである行についてはコスト値を「+ 2」する。この処理も、文字としてのバランスの面を考慮して行われる。隣接する行のフォント構成画素のパターンが同一であると、これまでのコスト計算においてはコスト値は同じになる。よって、そのコスト値を有する行が後述の拡大処理により拡大対象となると、隣接する行両方が拡大されることになる。その結果、同一の画素パターンを有する行が両方とも拡大されることになり、1つの文字中の一部分のみが必要以上に拡大されてしまうことになる。そこで、隣接する行が同一のパターンを有する場合は、コスト値を増加させて、それらの行が拡大されにくくすることにより、1つの文字中の特定箇所のみが偏って拡大されることを防止する。

【 0 0 5 3 】

こうして、コスト計算処理が終了すると、処理は図 3 に示す行方向拡大処理へ戻る。そして、CPU 1 は対象となるフォントの全ての行についてコスト計算が完了したか否かを判定し（ステップ S 1 3）、完了していない場合は全ての行のコスト計算を行う。全ての行についてコスト計算が完了した場合（ステップ S 1 3 ; Yes）、CPU 1 は図 7（a）に示すように、コストの低い順に行を順序付けする（ステップ S 1 4）。図 7（a）においては、コストの小さい行から順に、A、B、C、…と順序が付されている。なお、同じコストを有する行が複数ある場合は、文字の中央に近い行から順に上位の順序を付すことにより、拡大する方向が左右方向において偏らないようにする。図 7（a）の例では、コストが「3」である列が 6 つあるが、それらについて中央から順に左右に分散するように C、D、E、F、G、H と順序付けをしている。こうして、文字の中央から左右にバランスよく拡大がなされるように、拡大対象となる順序を付けている。

【 0 0 5 4 】

コストの低い順に各行の順序付けが終了すると、CPU 1 は、拡大率に基づいて拡大する行の数を決定する（ステップ S 1 5）。ここで、拡大率とは、今回のフォント拡大処理により、ビットマップフォントの元データを拡大する割合を指す。ステップ S 1 においてフォント拡大指示がなされる場合、その指示には、拡大率の情報が含まれている。例えば、ユーザが文字を見やすくするために文字の拡大を指示する場合、その拡大率（例えば、1 2 0 %、1 5 0 % など）を選択することになる。また、ユーザの指示に拘わらず、携帯端末装置 1 0 の特定の処理によって自動的にフォント拡大処理がなされる場合でも、必ず拡大率は決定される。よって、その拡大率に基づいて、対象となるフォントの行数のうち、いくつの行を拡大するかを決定する。例えば、拡大の対象となるフォントの行数が 1 5 行（1 5 ドット）であり、拡大率が 1 2 0 %（1. 2 倍）と決定された場合、拡大する行の数は $1 5 \times 1. 2 = 1 8$ （行）となり、3 つの行だけ拡大することになる。図 7（a）及び（b）の例はこの場合の例である。

【 0 0 5 5 】

こうして、拡大する行の数が決定すると、CPU 1 は、拡大処理を実行する（ステップ S 1 6）。即ち、コストが小さい行から順に、拡大する行の数だけ拡大

を行う。この場合の拡大は、例えば、拡大の対象となる行と同一のフォント画素パターンの行を拡大の対象となる行の隣に挿入することにより行われる。図 7 (a) 及び (b) の例では、前述のように拡大する行の数は 3 行であるので、図 7 (a) に示すコストの小さい行から順に 3 つの行 (コスト順序が「A」、「B」、「C」の行) について、同一のフォント構成画素パターンを有する行を挿入する。その結果、図 7 (b) に示すように、3 つの行が挿入されて、全体として横方向に 18 行 (18 ドット) の拡大文字が得られている。

【 0 0 5 6 】

こうして、行方向拡大処理が終了すると、処理は図 2 に示すメインルーチンへ戻る。そして、CPU 1 は、列方向拡大処理を行う (ステップ S 4)。なお、この列方向拡大処理において対象となるフォントは、行方向への拡大処理後のフォントとなる。よって、図 7 (a) 及び (b) の例では、横方向が 18 行 (ドット) となったフォントに対して行われる。

【 0 0 5 7 】

列方向拡大処理は、基本的に図 3 に示す行方向拡大処理と同様の処理を、行毎ではなく、列ごとに行う。即ち、拡大対象となるフォントを列ごとに分割し (ステップ S 1 1)、各列についてコスト計算を行い (ステップ S 1 2)、コストの低い順に各列について順序付けを行い (ステップ S 1 4)、拡大率に基づいて拡大対象となる列の数を決定し (ステップ S 1 5)、対象となる列に拡大処理を実行する (ステップ S 1 6)。なお、図 4 に示すコスト計算処理においては、縦方向に隣接画素を考慮したコストの重み付け (ステップ S 2 2)、スペースを考慮した重み付け (ステップ S 2 3)、及び同一コストを有する列の重み付け (ステップ S 2 4) を実行する。また、本例の拡大処理では、対象となるフォントを縦方向及び横方向に同一の拡大率で拡大するので、ステップ S 1 5 で使用する拡大率は行方向拡大処理において使用する拡大率と同一となる。

【 0 0 5 8 】

こうして、行方向及び列方向に拡大処理が完了すると、CPU 1 は得られたフォント、即ち処理フォントを処理フォントメモリ 1 4 に記憶し (ステップ S 5)、必要に応じて表示部 1 2 に表示する。こうして、拡大後のフォントが携帯端末

装置 1 0 の表示部 1 2 上に表示される。

【 0 0 5 9 】

以上のように、本発明のフォント拡大処理では、対象となるフォントの構成（フォント構成画素のパターン）に基づいて行、列ごとにコストを計算し、コストに基づいて適切な行、列を拡大する。具体的には、線分に近い行、列は拡大されにくくすることにより、文字中の線分部分が必要以上に拡大されることを防止する（ステップ S 2 2）。また、端部にスペースが多い文字については、スペース部分のみが拡大されて文字部分の拡大が不十分になることを防止する（ステップ S 2 3）。さらに、同一パターンを有する行、列が隣接する場合には、それらが拡大されにくくして、文字中の特定の部分のみが必要以上に拡大されることを防止する（ステップ S 2 4）。これらの処理により、文字を構成する画素パターンを考慮して、拡大後の文字のバランスが不自然とならないようにし、自然な拡大を可能としている。

【 0 0 6 0 】

なお、以上説明したフォント拡大処理は、表示部 1 2 上に表示すべき文字列が決定された時点で 1 文字ずつ実行することができる。また、ユーザが表示フォントのサイズ変更などを指定した場合には、フォント ROM 2 2 内に予め用意されている全ての元データについて拡大処理を行ってその結果を処理フォントメモリ 1 4 内に格納してもよい。いずれの場合でも、上述のようにフォント拡大処理自体は整数の単純な演算で済むので、処理に必要な時間は非常に短く、ユーザが不快に感じるほどの処理時間を要することはない。

【 0 0 6 1 】

〔フォント縮小処理〕

次に、フォント縮小処理について説明する。フォント縮小処理の図 8 に示す。フォント縮小処理は、基本的にフォント拡大処理と同様に行われる。即ち、CPU 1 は、フォント縮小指示を受け取ると（ステップ S 4 1）、対象となるフォントの元データをフォント ROM 2 2 から抽出し、作業メモリである RAM 2 4 に展開する（ステップ S 4 2）。次に、CPU 1 は、行方向縮小処理を行い（ステップ S 4 3）、さらに列方向縮小処理を行う（ステップ S 4 4）。そして、処理

フォントが作成されると、CPU 1 は、作成された処理フォントを処理フォントメモリ 1 4 に記憶する（ステップ S 4 5）とともに、表示部 1 2 に表示する（ステップ S 4 6）。こうして、フォント縮小処理がなされる。

【0062】

次に、フォント縮小処理における行方向縮小処理について図 9 を参照して説明する。図 9 は、行方向縮小処理のフローチャートである。図 9 に示す行方向縮小処理は、基本的に図 3 に示すフォント拡大処理における行方向拡大処理と同様の手順で行われる。即ち、CPU 1 は、対象となるフォントを行単位に分割し（ステップ S 5 1）、行ごとにコストを算出し（ステップ S 5 2）、全ての行についてコストが得られると（ステップ S 5 3；Yes）、コストの低い順に行を順序付ける（ステップ S 5 4）。次に、CPU 1 は、縮小率に基づいて縮小する行の数を決定し（ステップ S 5 5）、対象となる行について縮小処理を行う。

【0063】

このように、フォント縮小処理における行方向縮小処理は、基本的な流れはフォント拡大処理における行方向拡大処理と同様である。但し、フォント縮小処理における行方向縮小処理は、ステップ S 5 2 におけるコスト計算方法及びステップ S 5 6 における縮小方法において、フォント拡大処理の場合と異なる。以下、これについて説明する。

【0064】

まず、コスト計算について説明する。フォント拡大処理におけるコストは、線分量、即ち各行又は列に含まれる画素が線分に近いか、ドットに近いかを示す値であった。これに対し、フォント縮小処理におけるコストは、類似度、即ち各行又は列が、隣接する行又は列にどの程度類似しているかを示す値である。具体的には、CPU 1 は各行に対し、隣の行との排他的論理和（XOR）を算出する。排他的論理和の値は、隣接する画素が同じであると「0」となり、隣接する画素が異なると「1」となる。よって、各行に含まれる各フォント構成画素について排他的論理和を求め、その値が「1」となるフォント構成要素の数をその行のコストとする。よって、コストが高いほど、その行と隣接する行とのフォント画素パターンは非類似であり、コストが低いほど、その行と隣接する行とのフォント

画素パターンは類似であることになる。こうして、コストは隣接する行との類似度（厳密には、非類似度）を示す値となる。

【 0 0 6 5 】

即ち、ある行のコストが低いということは、その行のフォント画素パターンが隣接する行のフォント画素パターンと類似していることを意味する。よって、類似している画素パターンを有する行を優先的に縮小の対象とすることにより、類似している行を削除する。これは、隣接する行が相互に類似している場合には、それを優先的に削除しても、文字全体に与える影響、違和感は少ないであろうとの発想に基づいている。これにより、自然な縮小が可能となる。

【 0 0 6 6 】

行方向縮小処理が終わると、次に列方向縮小処理が行われるが（ステップ S 4 4）、これは行方向縮小処理と基本的に同様である。対象となるフォントを列単位に分割し（ステップ S 5 1）、列ごとにコストを算出し（ステップ S 5 2）、全ての列についてコストが得られると（ステップ S 5 3 ; Yes）、コストの低い順に列を順序付けする（ステップ S 5 4）。そして、CPU 1 は、縮小率に基づいて縮小する列の数を決定し（ステップ S 5 5）、対象となる列について縮小処理を行う。

【 0 0 6 7 】

図 1 0（a）～（c）に列方向縮小処理の様子を示す。まず、図 1 0（a）に示すように各列についてコスト計算が行われ、次に図 1 0（b）に示すようにコストの低い順に列の順序付けが行われ、コストが低い列から順に縮小の対象となる。ステップ S 5 6 の縮小処理においては、例えば図 1 0（c）に示すように、コストの低い列から順にその行が削除される。図 1 0（c）の例では、コストの低い 3 つの列（A、B、C）が削除されている。

【 0 0 6 8 】

以上説明したように、本発明のフォント縮小処理では、対象となるフォントを行方向及び列方向に分割し、それぞれ隣接する行又は列との類似度を考慮して縮小を行う。よって、フォント画素パターンが類似する行又は列が隣接している部分を優先して縮小（行又は列の削除）を行うので、縮小後のフォントが不自然と

なることが少なくなる。

【 0 0 6 9 】

図 1 1 (a) 及び (b) に、単純な座標変換によるフォント拡大・縮小処理と、本発明によるフォント拡大・縮小処理により得られた拡大・縮小フォントの例を示す。図からわかるように、単純な座標変換処理による拡大・縮小フォントでは縦や横のストロークが拡大により 2 重になったり（「本」という文字の横線、「肌」という文字の右側の縦線など）、斜めの線がギザギザに見えたりしている。また、縮小により、文字の左右のバランスが崩れたり（「本」という文字の左右のバランス）、画素同士が連結して別の文字に見えたり（「肌」という文字の左側が「月」ではなく「目」に見える）している。これに対し、本発明によるフォント拡大・縮小処理では、そのような不具合は生じず、自然な拡大・縮小フォントが得られている。

【 0 0 7 0 】

〔行方向処理と列方向処理の順序〕

次に、フォント拡大／縮小処理における行方向拡大／縮小処理と、列方向拡大／縮小処理の順序について検討する。先に説明したフォント拡大／縮小処理においては、まず行方向の拡大／縮小処理を行い、次に列方向の拡大／縮小処理を行っている（即ち、まず横方向に拡大／縮小し、次に縦方向に拡大／縮小している）。しかし、これは 1 つの例であり、列方向の拡大／縮小処理を先に行い、次に行方向の拡大／縮小処理を行うことも可能である。そして、いずれが好ましいかは、処理の対象となるフォントの構成に依存する。

【 0 0 7 1 】

図 1 2 (a) 及び (b) に、同一の画素パターンに対して異なる順序で拡大を行った場合の例を示す。図 1 2 (a) の例は、まず縦方向に拡大を行い、次に横方向に拡大を行った例である。図 1 2 (a) において、最も左の元データ 5 0 a において水平方向のコストを計算すると上から 2 行目が最も小さい。よって、上から 2 行目を縦方向に 1 画素分拡大すると真ん中のデータ 5 0 b が得られる。次に、データ 5 0 b において垂直方向のコストを計算すると左から 4 行目及び 5 行目のコストが最も小さい。よって、この一方を横方向に 1 画素分拡大すると、最

も右の拡大後データ 5 0 c が得られる。

【 0 0 7 2 】

一方、図 1 2 (b) の例は、まず横方向に拡大を行い、次に縦方向に拡大を行った例である。図 1 2 (b) において、最も左の元データ 6 0 a において垂直方向のコストを計算すると、左から 4 行目及び 5 行目が最も小さい。よって、その一方を横方向に 1 画素分拡大すると真ん中のデータ 6 0 b が得られる。次に、データ 6 0 b において水平方向のコストを計算すると、上から 1 行目及び 3 行目のコストが最も小さい。よって、上から 1 行目を縦方向に 1 画素分拡大すると、最も右の拡大後データ 6 0 c が得られる。

【 0 0 7 3 】

このように、元データが同一でも、縦→横の順で拡大するか、横→縦の順で拡大するかによって、結果として得られるデータは異なる。図 1 2 (a) 及び (b) の例では、元データに対しては図 1 2 (a) に示すように縦→横の順で拡大した方が自然に拡大がなされる。よって、全ての文字に対して自然な拡大を行うためには、文字ごとに実験的に縦→横、及び横→縦の順で拡大処理を行ってどちらの順序で拡大を行う方が自然な処理フォントが得られるかを予め決めておくことが好ましい。そして、図 1 2 (c) に模式的に示すように、各フォントの元データに対して、拡大／縮小処理を行う場合に、縦→横の順がよいか、横→縦の順がよいかを示す処理順序情報（例えばフラグなど）を属性データとして付属させることがより好ましい。こうすれば、CPU 1 6 は、フォント拡大／縮小処理の実行時にフォント ROM 2 2 から読み出したフォントに付属する処理順序情報を参照し、それに示す順序でフォントの拡大・縮小を行えばよい。なお、同一のフォントであっても、拡大の場合に好ましい処理順序と、縮小の場合に好ましい処理順序とが異なる場合もあるので、拡大／縮小について個別に処理順序情報を用意することが好ましい。さらに、拡大率／縮小率によって好ましい処理順序が異なる場合には、拡大率／縮小率ごとに処理順序情報を用意すればよい。

【 0 0 7 4 】

なお、本実施形態においては、コストの小さい行又は列から順に画素の拡大又は縮小処理を行うものとしたが、本発明の適用範囲はこれに限られるものではない。

く、例えば本実施形態において用いたコストの逆数を取ったものを新しい意味でコストと定義すれば、コストの大きい行又は列から順に画素の拡大又は縮小処理を行うことにより、本実施形態と同様の効果を得ることができる。

【図面の簡単な説明】

【図 1】

本発明の実施形態にかかるビットマップフォントの拡大・縮小処理を適用した携帯端末装置の概略構成を示す。

【図 2】

本発明によるフォント拡大処理のフローチャートである。

【図 3】

行（列）方向拡大処理のフローチャートである。

【図 4】

フォント拡大処理におけるコスト計算処理のフローチャートである。

【図 5】

フォント拡大処理におけるコスト計算処理例を示す図である。

【図 6】

コスト計算処理の例を示す図である。

【図 7】

行（列）方向拡大処理の例を示す図である。

【図 8】

フォント縮小処理のフローチャートである。

【図 9】

行（列）方向縮小処理のフローチャートである。

【図 1 0】

フォント縮小処理におけるコスト計算処理例を示す図である。

【図 1 1】

本発明によるフォント拡大／縮小処理と、単純な座標変換によるフォント拡大／縮小処理の結果を比較する図である。

【図 1 2】

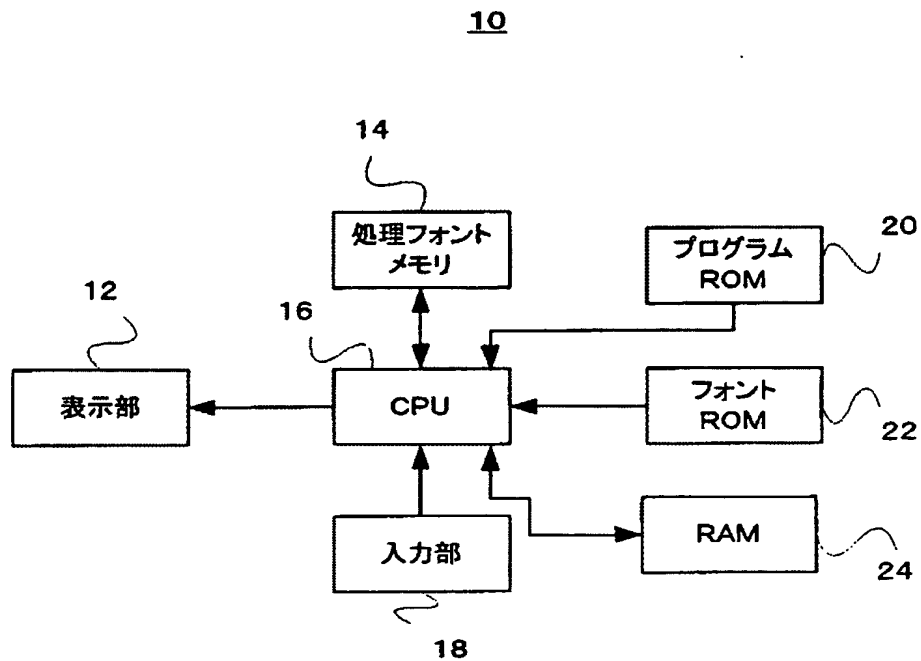
行方向処理と列方向処理の処理順序と処理結果との比較、及び、処理順序情報を含むフォントデータのデータ構造例を示す図である。

【符号の説明】

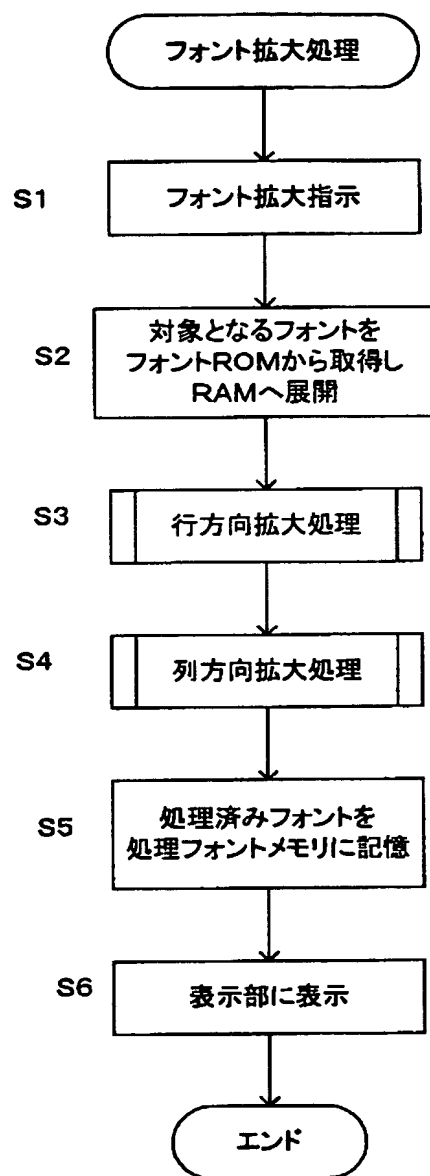
- 1 0 携帯端末装置
- 1 2 表示部
- 1 4 処理フォントメモリ
- 1 6 C P U
- 1 8 入力部
- 2 0 プログラム R O M
- 2 2 フォント R O M
- 2 4 R A M

【書類名】 図面

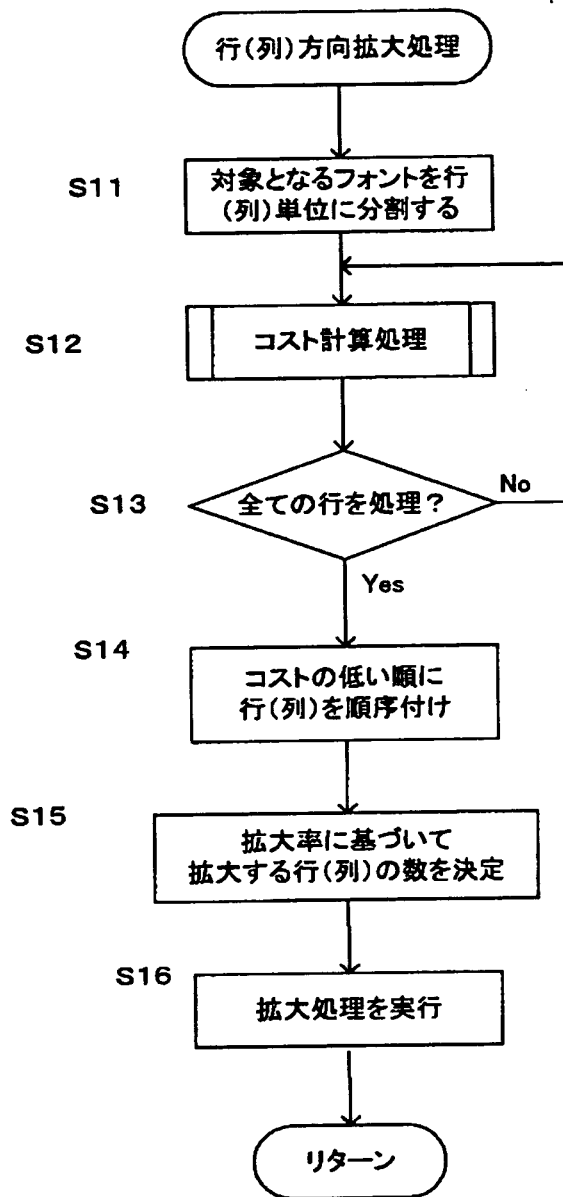
【図 1】



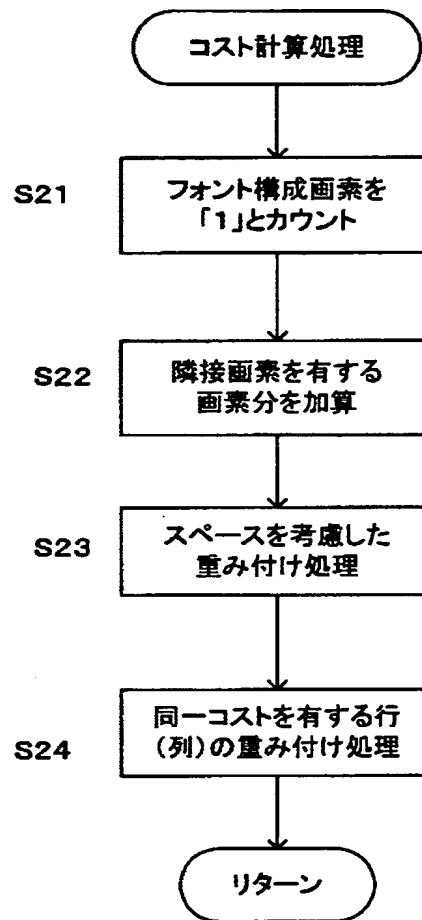
【図 2】



【図 3】

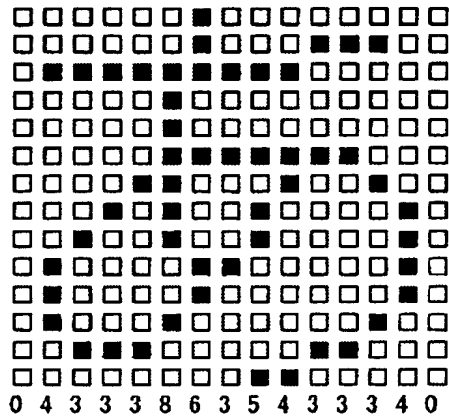


【図 4】



【図 5】

S21: フォント構成画素(■)をカウントする。

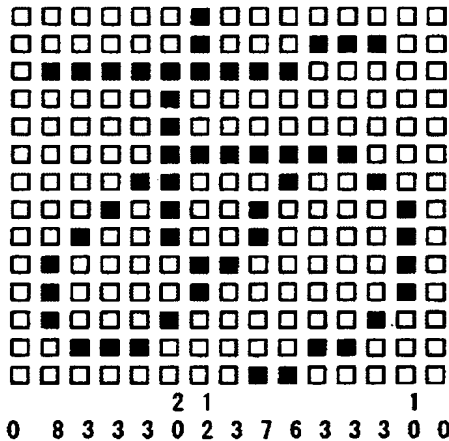


左の例では↓方向に
■をカウントしている。

← 各行のコスト (一の位)

(a)

S22: 隣接する画素分を加算
(連続する■は、さらに+2する)。



□■□□□ = $2 + 2 = 4$ となり
■□□□■ = 3 よりもコストが高い。

□■□■□ = $4 + 2 + 2 + 2 = 10$ となり、
■□□□■ = $(2 + 2) + (2 + 2) = 8$
よりもコストが高い。

← 各行のコスト (十の位)

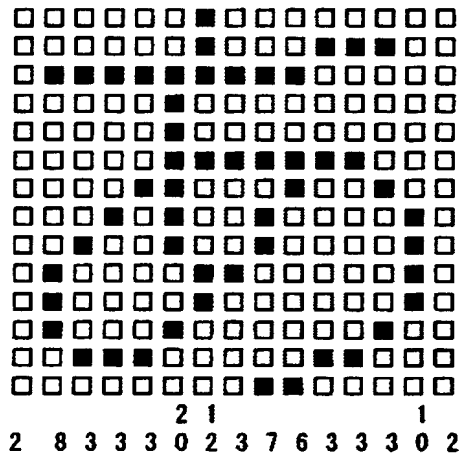
← 各行のコスト (一の位)

(b)

【図 6】

S 2 3 : スペースを考慮した重み付け処理

両端およびその 1 画素内側がスペース (■がゼロ) なら、
それぞれ + 2、+ 1 する。

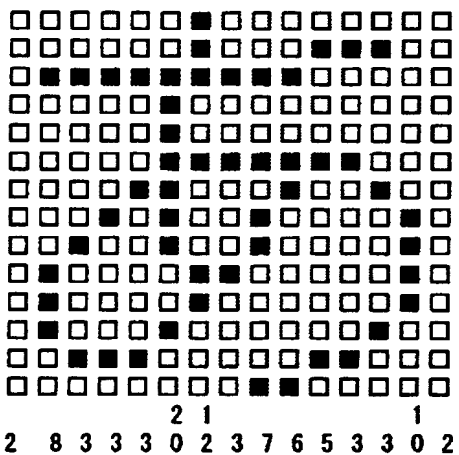


各行のコスト (十の位)
各行のコスト (一の位)

(a)

S 2 4 : 同一コストを有する行 (列) の重み付け処理

右側の列とパターンが同じなら、+ 2 する。

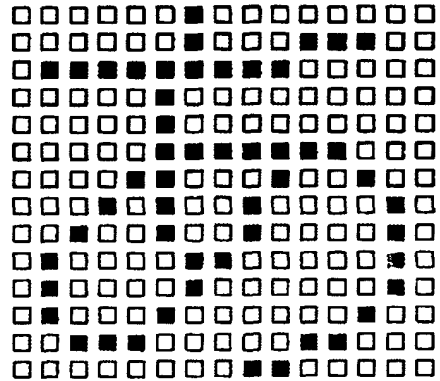


拡大する箇所が偏らないように
重み付けを行う。

(b)

【図 7】

S14: コストが低い順に文字の中心から外側に向かって順序付け



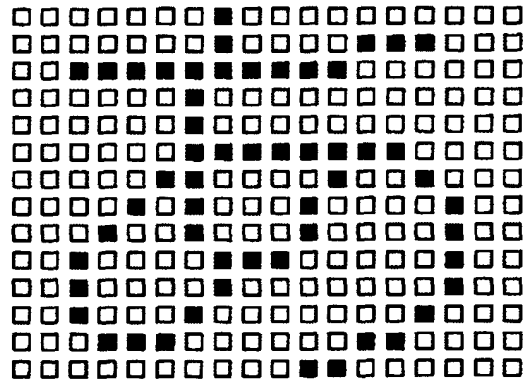
拡大する方向が偏らないように、
分散させる。

2 1 1
2 8 3 3 3 0 2 3 7 6 5 3 3 0 2
B L G E D O N C K J I F H M A

各行のコスト（十の位）
各行のコスト（一の位）
コストの低い順（A、B、...）

(a)

S16: 拡大処理



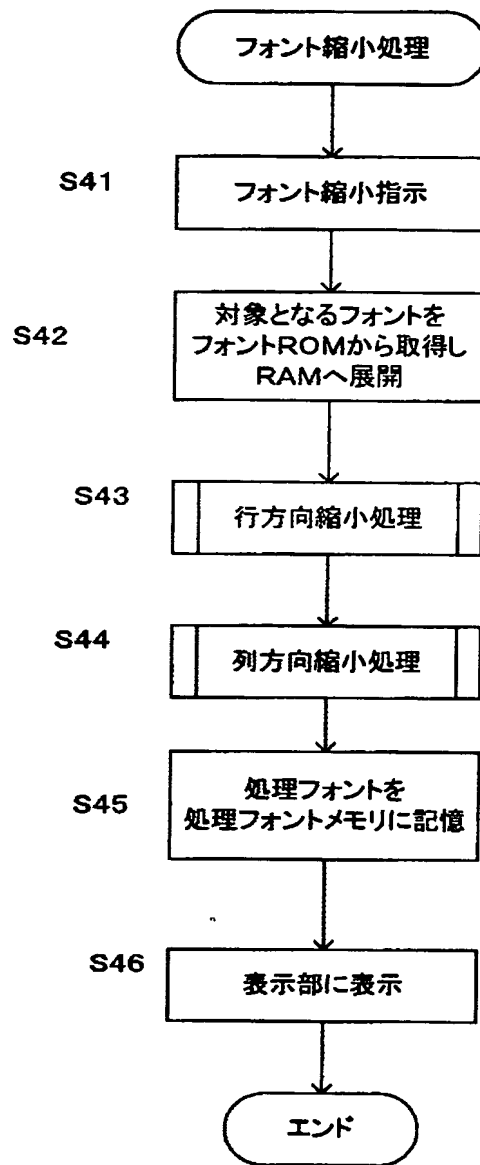
左の例では、横方向に3ドット拡大
している（15→18）。

2 1 1
2 8 3 3 3 0 2 3 7 6 5 3 3 0 2
B L G E D O N C K J I F H M A

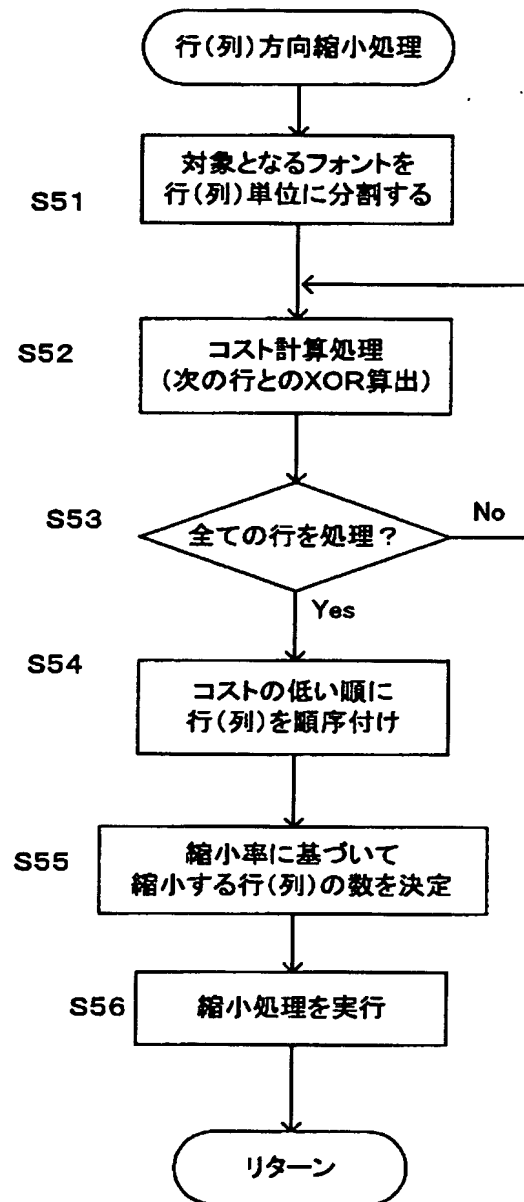
各行のコスト（十の位）
各行のコスト（一の位）
コストの低い順（A、B、...）

(b)

【図 8】

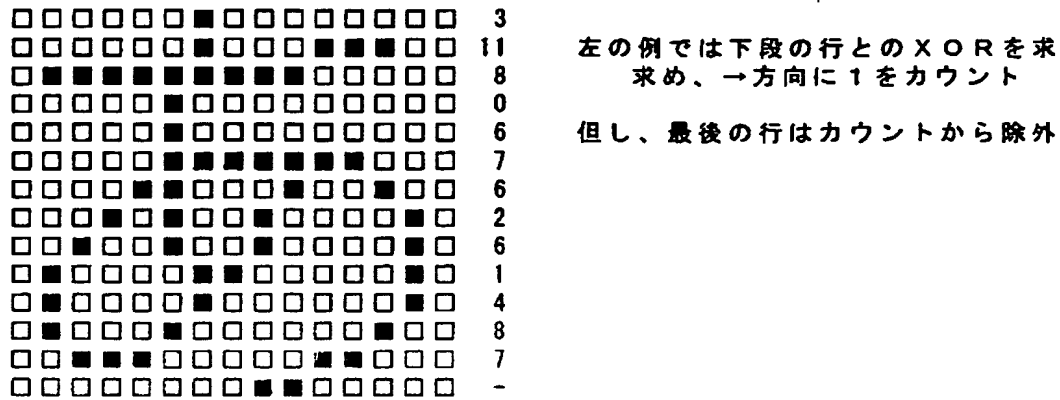


【図 9】



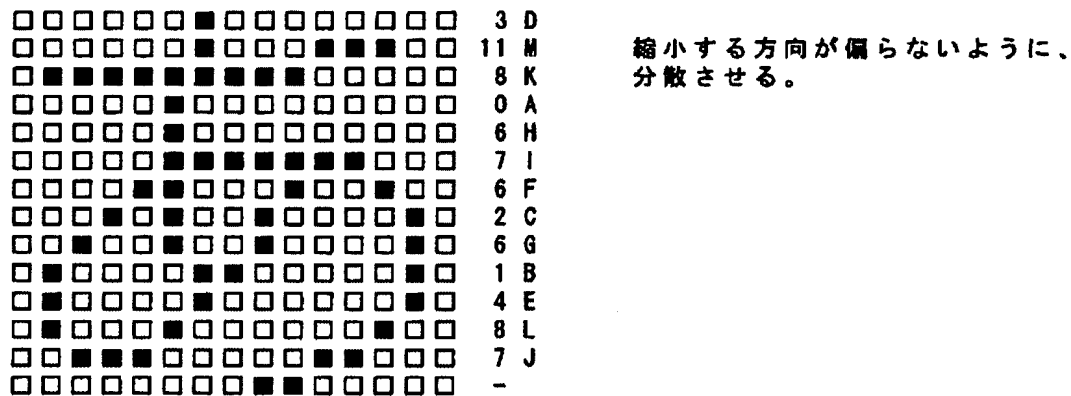
【図 1 0】

S52: 隣の列との排他的論理和 (XOR) を求め、XOR が 1 となる数を数える



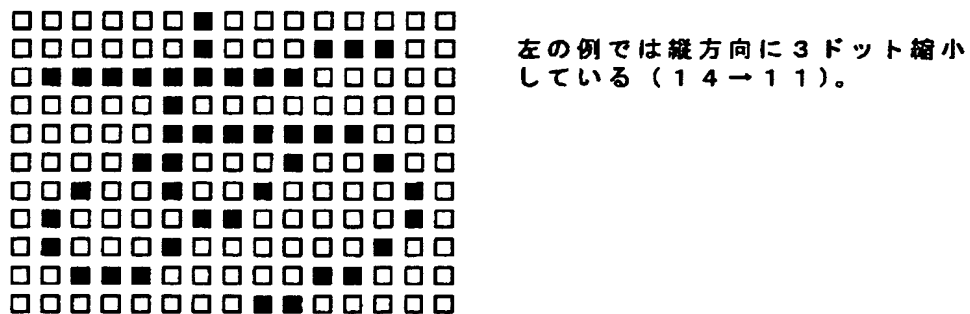
(a)

S54: コストが低い順に文字の中心から外側に向かって順序付け



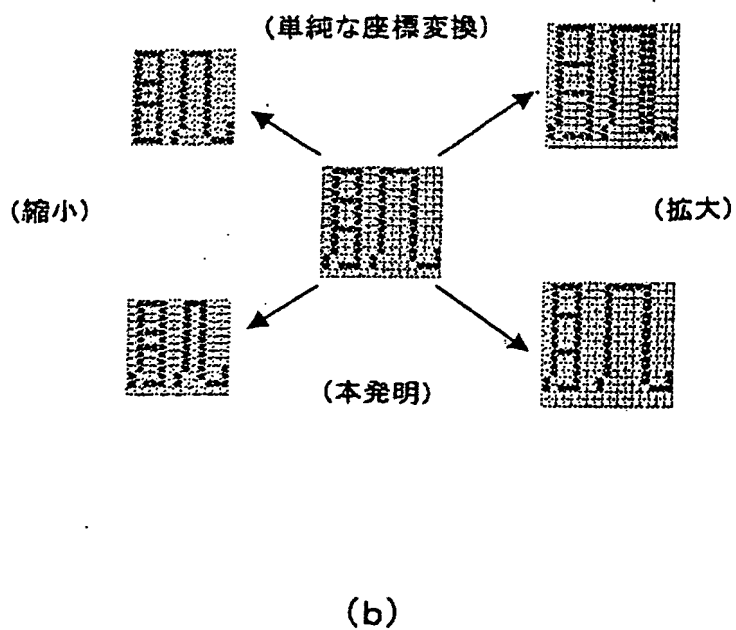
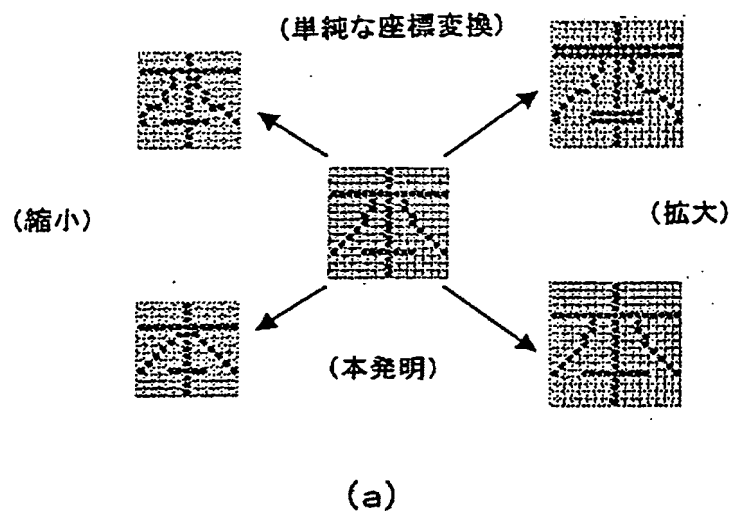
(b)

S56: 縮小処理

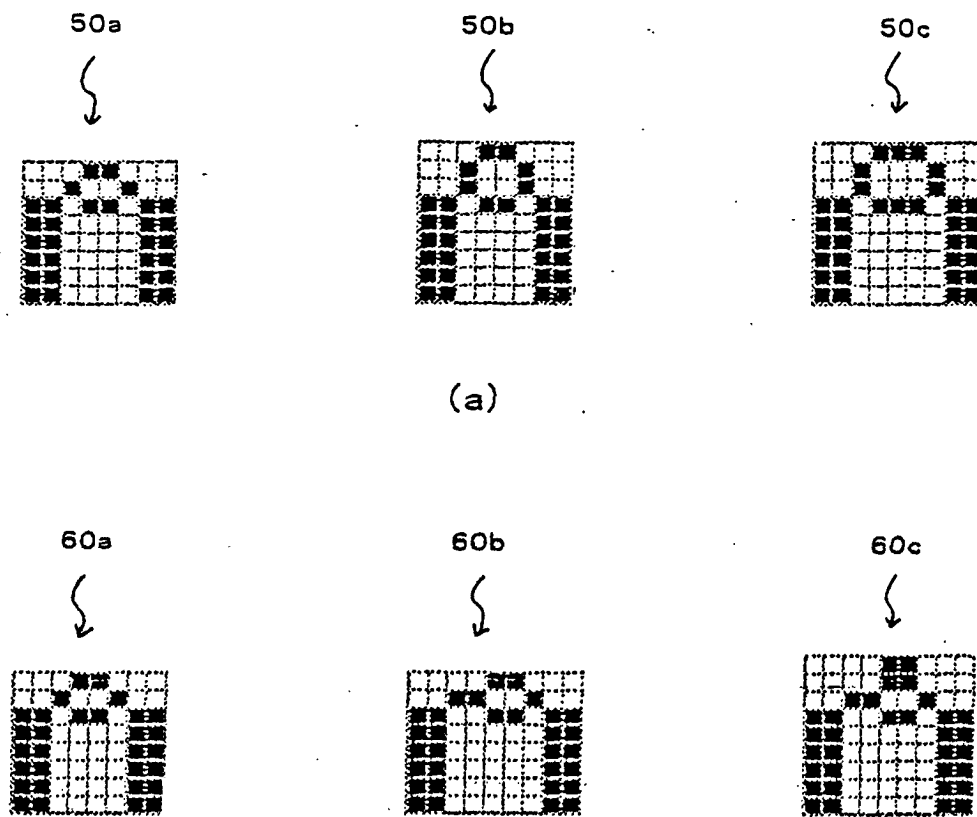


(c)

【図 1 1】

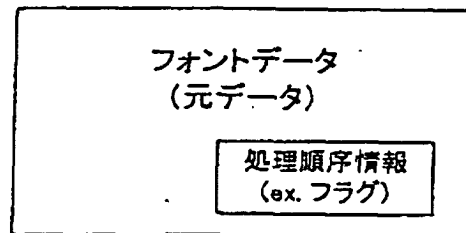


【図 12】



(a)

(b)



(c)

【書類名】 要約書

【要約】

【課題】 携帯電話やPDAなどで使用するサイズの小さなビットマップフォントを自然に拡大・縮小する。

【解決手段】 拡大・縮小の対象となるフォントを行、列に分割し、行、列毎の構成画素パターンに基づいてコストを計算する。拡大処理においてコストは線分量を示し、コストの小さい行又は列、即ち線分に近い画素配列を有する行又は列が拡大されにくくする。また、縮小処理においてコストは隣接する行、列との類似度を示し、近傍に類似する画素パターンがある行又は列は優先的に縮小の対象となる。実際に拡大、縮小の対象となるフォントの画素構成に基づいて、拡大、縮小の対象とする行、列を決定するので、自然な拡大・縮小フォントが得られる。

【選択図】 図1

認定・付加情報

特許出願の番号	特願2002-255257
受付番号	50201300846
書類名	特許願
担当官	第一担当上席 0090
作成日	平成14年 9月 2日

<認定情報・付加情報>

【提出日】	平成14年 8月30日
-------	-------------

出 願 人 履 歴 情 報

識別番号 [000002369]

1. 変更年月日 1990年 8月20日
[変更理由] 新規登録
住 所 東京都新宿区西新宿2丁目4番1号
氏 名 セイコーエプソン株式会社